Matrix Data: Clustering: Part 2

Instructor: Yizhou Sun
yzsun@ccs.neu.edu

October 19, 2014
# Methods to Learn

<table>
<thead>
<tr>
<th></th>
<th>Matrix Data</th>
<th>Set Data</th>
<th>Sequence Data</th>
<th>Time Series</th>
<th>Graph &amp; Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>Decision Tree; Naïve Bayes; Logistic Regression; SVM; kNN</td>
<td></td>
<td>HMM</td>
<td></td>
<td>Label Propagation</td>
</tr>
<tr>
<td>Clustering</td>
<td>K-means; hierarchical clustering; DBSCAN; <strong>Mixture Models; kernel k-means</strong></td>
<td></td>
<td></td>
<td></td>
<td>SCAN; Spectral Clustering</td>
</tr>
<tr>
<td>Frequent Pattern Mining</td>
<td>Apriori; FP-growth</td>
<td><strong>GSP; PrefixSpan</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prediction</td>
<td>Linear Regression</td>
<td></td>
<td></td>
<td></td>
<td>Autoregression</td>
</tr>
<tr>
<td>Similarity Search</td>
<td></td>
<td></td>
<td></td>
<td><strong>DTW</strong></td>
<td><strong>P-PageRank</strong></td>
</tr>
<tr>
<td>Ranking</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>PageRank</strong></td>
</tr>
</tbody>
</table>
Matrix Data: Clustering: Part 2

- Revisit K-means
- Mixture Model and EM algorithm
- Kernel K-means
- Summary
Recall K-Means

• Objective function
  
  \[ J = \sum_{j=1}^{k} \sum_{c(i)=j} ||x_i - c_j||^2 \]

• Total within-cluster variance

• Re-arrange the objective function
  
  \[ J = \sum_{j=1}^{k} \sum_i w_{ij} ||x_i - c_j||^2 \]

  • \( w_{ij} \in \{0,1\} \)
  
  • \( w_{ij} = 1, \text{if } x_i \text{ belongs to cluster } j; w_{ij} = 0, \text{otherswise} \)

• Looking for:
  
  • The best assignment \( w_{ij} \)
  
  • The best center \( c_j \)
Solution of K-Means

\[ J = \sum_{j=1}^{k} \sum_{i} w_{ij} ||x_i - c_j||^2 \]

• Iterations

  • **Step 1:** Fix centers \( c_j \), find assignment \( w_{ij} \) that minimizes \( J \)
    
    • \( \Rightarrow w_{ij} = 1 \), if \( ||x_i - c_j||^2 \) is the smallest

  • **Step 2:** Fix assignment \( w_{ij} \), find centers that minimize \( J \)
    
    • \( \Rightarrow \) first derivative of \( J = 0 \)
    
    • \( \Rightarrow \frac{\partial J}{\partial c_j} = -2 \sum_i w_{ij} (x_i - c_j) = 0 \)

    • \( \Rightarrow c_j = \frac{\sum_i w_{ij} x_i}{\sum_i w_{ij}} \)

    • Note \( \sum_i w_{ij} \) is the total number of objects in cluster \( j \)
Converges! Why?
Limitations of K-Means

• K-means has problems when clusters are of differing
  • Sizes
  • Densities
  • Non-Spherical Shapes
Limitations of K-Means: Different Density and Size

Original Points

K-means (3 Clusters)
Limitations of K-Means: Non-Spherical Shapes

Original Points  K-means (2 Clusters)
Demo

Connections of K-means to Other Methods

- K-means
- Gaussian Mixture Model
- Kernel K-means
Matrix Data: Clustering: Part 2

• Revisit K-means

• Mixture Model and EM algorithm

• Kernel K-means

• Summary
Fuzzy Set and Fuzzy Cluster

• Clustering methods discussed so far
  • Every data object is assigned to exactly one cluster

• Some applications may need for fuzzy or soft cluster assignment
  • Ex. An e-game could belong to both entertainment and software

• Methods: fuzzy clusters and probabilistic model-based clusters

• Fuzzy cluster: A fuzzy set $S: F_S : X \rightarrow [0, 1]$ (value between 0 and 1)
Cluster analysis is to find hidden categories. A hidden category (i.e., *probabilistic cluster*) is a distribution over the data space, which can be mathematically represented using a probability density function (or distribution function).

- **Ex.** categories for digital cameras sold
  - consumer line vs. professional line
  - density functions $f_1, f_2$ for $C_1, C_2$
  - obtained by probabilistic clustering

- A **mixture model** assumes that a set of observed objects is a mixture of instances from multiple probabilistic clusters, and conceptually each observed object is generated independently.

- **Our task:** infer a set of $k$ probabilistic clusters that is mostly likely to generate $D$ using the above data generation process.
Mixture Model-Based Clustering

- A set $C$ of $k$ probabilistic clusters $C_1, \ldots, C_k$ with probability density functions $f_1, \ldots, f_k$, respectively, and their probabilities $w_1, \ldots, w_k$, $\sum_j w_j = 1$

- Probability of an object $i$ generated by cluster $C_j$ is: $P(x_i, z_i = C_j) = w_j f_j(x_i)$

- Probability of $i$ generated by the set of cluster $C$ is: $P(x_i) = \sum_j w_j f_j(x_i)$
Maximum Likelihood Estimation

• Since objects are assumed to be generated independently, for a data set $D = \{x_1, \ldots, x_n\}$, we have,

$$P(D) = \prod_i P(x_i) = \prod_i \sum_j w_j f_j(x_i)$$

• Task: Find a set $C$ of $k$ probabilistic clusters s.t. $P(D)$ is maximized
The EM (Expectation Maximization) Algorithm

• The (EM) algorithm: A framework to approach maximum likelihood or maximum a posteriori estimates of parameters in statistical models.

• **E-step** assigns objects to clusters according to the current fuzzy clustering or parameters of probabilistic clusters

  \[ w_{ij}^t = p(z_i = j | \theta_j^t, x_i) \propto p(x_i | C_j^t, \theta_j^t)p(C_j^t) \]

• **M-step** finds the new clustering or parameters that maximize the expected likelihood
Case 1: Gaussian Mixture Model

• Generative model
  • For each object:
    • Pick its distribution component:
      \[ Z \sim \text{Multi}(w_1, \ldots, w_k) \]
    • Sample a value from the selected distribution:
      \[ X \sim N(\mu_Z, \sigma_Z^2) \]
  • Overall likelihood function
    \[ L(D | \theta) = \prod_i \sum_j w_j p(x_i | \mu_j, \sigma_j^2) \]
• Q: What is \( \theta \) here?
Estimating Parameters

- \( L(D; \theta) = \sum_i \log \sum_j w_j p(x_i | \mu_j, \sigma_j^2) \)

- Considering the first derivative of \( \mu_j \):

\[
\frac{\partial L}{\partial u_j} = \sum_i \frac{w_j}{\sum_j w_j p(x_i | \mu_j, \sigma_j^2)} \frac{\partial p(x_i | \mu_j, \sigma_j^2)}{\partial \mu_j}
\]

\[
= \sum_i \frac{w_j p(x_i | \mu_j, \sigma_j^2)}{\sum_j w_j p(x_i | \mu_j, \sigma_j^2)} \frac{1}{p(x_i | \mu_j, \sigma_j^2)} \frac{\partial p(x_i | \mu_j, \sigma_j^2)}{\partial \mu_j}
\]

\[
= \sum_i \frac{w_j p(x_i | \mu_j, \sigma_j^2)}{\sum_j w_j p(x_i | \mu_j, \sigma_j^2)} \frac{\partial \log p(x_i | \mu_j, \sigma_j^2)}{\partial u_j}
\]

\[ w_{ij} = P(Z = j | X = x_i, \theta) \]

Intractable!

Like weighted likelihood estimation; But the weight is determined by the parameters!
Apply EM algorithm

- An iterative algorithm (at iteration t+1)
  - E(expectation)-step
    - Evaluate the weight $w_{ij}$ when $\mu_j, \sigma_j, w_j$ are given
    $$w_{ij}^t = \frac{w_j^tp(x_i|\mu_j^t, (\sigma_j^2)^t)}{\sum_j w_j^tp(x_i|\mu_j^t, (\sigma_j^2)^t)}$$
  - M(maximization)-step
    - Evaluate $\mu_j, \sigma_j, \omega_j$ when $w_{ij}$’s are given that maximize the weighted likelihood
    - It is equivalent to Gaussian distribution parameter estimation when each point has a weight belonging to each distribution
    $$\mu_j^{t+1} = \frac{\sum_i w_{ij}^tx_i}{\sum_i w_{ij}^t}; (\sigma_j^2)^{t+1} = \frac{\sum_i w_{ij}^t||x_i-\mu_j^t||^2}{\sum_i w_{ij}^t}; w_j^{t+1} \propto \sum_i w_{ij}^t$$
K-Means: A Special Case of Gaussian Mixture Model

- When each Gaussian component with covariance matrix $\sigma^2 I$
  - Soft K-means
  - $p(x_i | \mu_j, \sigma^2) \propto \exp\left\{-\frac{(x_i - \mu_j)^2}{\sigma^2}\right\}$

- When $\sigma^2 \to 0$
  - Soft assignment becomes hard assignment
  - $w_{ij} \to 1$, if $x_i$ is closest to $\mu_j$ (why?)
Case 2: Multinomial Mixture Model

• Generative model
  • For each object:
    • Pick its distribution component:
      \( Z \sim Multi(w_1, \ldots, w_k) \)
    • Sample a value from the selected distribution:
      \( X \sim Multi(\beta_{Z1}, \beta_{Z2}, \ldots, \beta_{Zm}) \)

• Overall likelihood function
  \[ L(D | \theta) = \prod_i \sum_j w_j p(x_i | \beta_j) \]
  • \( \sum_j w_j = 1; \sum_l \beta_{jl} = 1 \)
  • Q: What is \( \theta \) here?
Application: Document Clustering

- A vocabulary containing m words
- Each document i:
  - A m-dimensional vector: \((c_{i1}, c_{i2}, \ldots, c_{im})\)
  - \(c_{il}\) is the number of occurrence of word \(l\) appearing in document \(i\)
- Under unigram assumption
  - \(p(x_i | \beta_j) = \frac{(\sum m c_{il})!}{c_{i1}! \ldots c_{im}!} \beta_{j1}^{c_{i1}} \ldots \beta_{jm}^{c_{im}}\)

Length of document
Constant to all parameters
Example

<table>
<thead>
<tr>
<th>&quot;Genetics&quot;</th>
<th>&quot;Evolution&quot;</th>
<th>&quot;Disease&quot;</th>
<th>&quot;Computers&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>evolution</td>
<td>disease</td>
<td>computer</td>
</tr>
<tr>
<td>genome</td>
<td>evolutionary</td>
<td>host</td>
<td>models</td>
</tr>
<tr>
<td>dna</td>
<td>species</td>
<td>bacteria</td>
<td>information</td>
</tr>
<tr>
<td>genetic</td>
<td>organisms</td>
<td>diseases</td>
<td>data</td>
</tr>
<tr>
<td>genes</td>
<td>life</td>
<td>resistance</td>
<td>computers</td>
</tr>
<tr>
<td>sequence</td>
<td>origin</td>
<td>bacterial</td>
<td>system</td>
</tr>
<tr>
<td>gene</td>
<td>biology</td>
<td>new</td>
<td>network</td>
</tr>
<tr>
<td>molecular</td>
<td>groups</td>
<td>strains</td>
<td>systems</td>
</tr>
<tr>
<td>sequencing</td>
<td>phylogenetic</td>
<td>control</td>
<td>model</td>
</tr>
<tr>
<td>map</td>
<td>living</td>
<td>infectious</td>
<td>parallel</td>
</tr>
<tr>
<td>information</td>
<td>diversity</td>
<td>malaria</td>
<td>methods</td>
</tr>
<tr>
<td>genetics</td>
<td>group</td>
<td>parasite</td>
<td>networks</td>
</tr>
<tr>
<td>mapping</td>
<td>new</td>
<td>parasites</td>
<td>software</td>
</tr>
<tr>
<td>project</td>
<td>two</td>
<td>united</td>
<td>new</td>
</tr>
<tr>
<td>sequences</td>
<td>common</td>
<td>tuberculosis</td>
<td>simulations</td>
</tr>
</tbody>
</table>
Estimating Parameters

\[ l(D; \theta) = \sum_i \log \sum_j \omega_j \sum_l c_{il} \log \beta_{jl} \]

• Apply EM algorithm

  • **E-step:**
    \[ w_{ij} = \frac{w_j p(x_i|\beta_j)}{\sum_j w_j p(x_i|\beta_j)} \]

  • **M-step:** maximize weighted likelihood
    \[ \sum_i w_{ij} \sum_l c_{il} \log \beta_{jl} \]

\[ \beta_{jl} = \frac{\sum_i w_{ij} c_{il}}{\sum_{l'} \sum_i w_{ij} c_{il'}}; \omega_j \propto \sum_i w_{ij} \]

Weighted percentage of word l in cluster j
Better Way for Topic Modeling

• Topic: a word distribution
• Unigram multinomial mixture model
  • Once the topic of a document is decided, all its words are generated from that topic
• PLSA (probabilistic latent semantic analysis)
  • Every word of a document can be sampled from different topics
• LDA (Latent Dirichlet Allocation)
  • Assume priors on word distribution and/or document cluster distribution
Why EM Works?

- **E-Step:** computing a tight lower bound \( f \) of the original objective function at \( \theta_{old} \)
- **M-Step:** find \( \theta_{new} \) to maximize the lower bound

\[
\ell(\theta_{new}) \geq f(\theta_{new}) \geq f(\theta_{old}) = \ell(\theta_{old})
\]
**How to Find Tight Lower Bound?**

- \( \ell(\theta) = \log \sum_h p(d, h; \theta) \)
  - \( = \log \sum_h \frac{q(h)}{q(h)} p(d, h; \theta) \)
  - \( = \log \sum_h q(h) \frac{p(d, h; \theta)}{q(h)} \)

- **Jensen’s inequality**

- \( \log \sum_h q(h) \frac{p(d, h; \theta)}{q(h)} \geq \sum_h q(h) \log \frac{p(d, h; \theta)}{q(h)} \)

- **When “=” holds to get a tight lower bound?**
  - \( q(h) = p(h|d, \theta) \) (why?)
Advantages and Disadvantages of Mixture Models

• **Strength**
  - Mixture models are more general than partitioning
  - Clusters can be characterized by a small number of parameters
  - The results may satisfy the statistical assumptions of the generative models

• **Weakness**
  - Converge to local optimal (overcome: run multi-times w. random initialization)
  - Computationally expensive if the number of distributions is large, or the data set contains very few observed data points
  - Need large data sets
  - Hard to estimate the number of clusters
Matrix Data: Clustering: Part 2

• Revisit K-means

• Mixture Model and EM algorithm

• Kernel K-means

• Summary
Kernel K-Means

• How to cluster the following data?

• A non-linear map: \( \phi : R^n \rightarrow F \)
  • Map a data point into a higher/infinite dimensional space
  • \( x \rightarrow \phi(x) \)

• Dot product matrix \( K_{ij} \)
  • \( K_{ij} = \langle \phi(x_i), \phi(x_j) \rangle \)
Typical Kernel Functions

- Recall kernel SVM:

  \[ K(X_i, X_j) = (X_i \cdot X_j + 1)^h \]

  Polynomial kernel of degree \( h \):

  Gaussian radial basis function kernel:

  \[ K(X_i, X_j) = e^{-\|X_i - X_j\|^2 / 2\sigma^2} \]

  Sigmoid kernel:

  \[ K(X_i, X_j) = \tanh(\kappa X_i \cdot X_j - \delta) \]
Solution of Kernel K-Means

• Objective function under new feature space:

\[ J = \sum_{j=1}^{k} \sum_{i} w_{ij} \| \phi(x_i) - c_j \|^2 \]

• Algorithm

• By fixing assignment \( w_{ij} \)

\[ c_j = \frac{\sum_{i} w_{ij} \phi(x_i)}{\sum_{i} w_{ij}} \]

• In the assignment step, assign the data points to the closest center

\[ d(x_i, c_j) = \left\| \phi(x_i) - \frac{\sum_{i'} w_{i'j} \phi(x_{i'})}{\sum_{i'} w_{i'j}} \right\|^2 = \phi(x_i) \cdot \phi(x_i) - \]

\[ \frac{\sum_{i'} w_{i'j} \phi(x_i) \cdot \phi(x_{i'})}{\sum_{i'} w_{i'j}} + \frac{\sum_{i'} \sum_{l} w_{i'j} w_{lj} \phi(x_{i'}) \cdot \phi(x_l)}{(\sum_{i'} w_{i'j})^2} \]

Do not really need to know \( \phi(x) \), but only \( K_{ij} \)
Advantages and Disadvantages of Kernel K-Means

- **Advantages**
  - Algorithm is able to identify the non-linear structures.

- **Disadvantages**
  - Number of cluster centers need to be predefined.
  - Algorithm is complex in nature and time complexity is large.

- **References**
  - Kernel k-means and Spectral Clustering by Max Welling.
  - Kernel k-means, Spectral Clustering and Normalized Cut by Inderjit S. Dhillon, Yuqiang Guan and Brian Kulis.
  - An Introduction to kernel methods by Colin Campbell.
Matrix Data: Clustering: Part 2

- Revisit K-means
- Mixture Model and EM algorithm
- Kernel K-means
- Summary
Summary

• Revisit k-means
  • Derivative

• Mixture models
  • Gaussian mixture model; multinominal mixture model; EM algorithm; Connection to k-means

• Kernel k-means
  • Objective function; solution; connection to k-means